

PERSON
- Ssn : int  - Name : String  - MiddleName : String  - LastName : String
+ getSsn() : int  + getName() : String  + getMiddleName() : String  + getLastName() : String  + getFullName() : String

STUDENT
- StudentID : int  - StudentSemester : int  - StudentTranscript : TRANSCRIPT
+ getStudentID : int  + getStudentSemester() : int  + getStudentTranscript() : TRANSCRIPT

STAFF
- CoursesGiven : List<COURSE>  - AdvisesTo : List<STUDENT>
+ addCourse( Course COURSE) : void  + removeCourse ( Course COURSE) : void  + addStudent( Student STUDENT) : void  + removeStudent( Student STUDENT) : void

COURSE
- CourseID : String  - CourseName : String  - CoursePrerequisite : COURSE  - CourseType : String (NTE, TE etc.)  - CourseQuota : int  - CourseCredit : int  - CourseDate : List<String>  - EnrolledStudents : List<STUDENT>
+ getCourseID() : String  + getCourseName() : String  + getCoursePrerequisite() : COURSE  + getCourseType() : String  + getCourseQuota() : int  + getCourseCredit() : int  + getCourseDate : List<String>  + addStudent( Student STUDENT) : void  + removeStudent( Student STUDENT) : void

TAKEN_COURSE
- StudentID : int  - TakenCourseStatus : String (pass,fail,repeat,current)  - TakenCourseScore : float
+ getStudentID() : int  + getTakenCourseStatus() : String  + getTakenCourseScore() : float

CLASSROOM
- ClassroomID : String  - ClassroomQuota : int
+ getClassroomID : String  + getClassroomCuota : int

SELECTION_PROBLEM
- ProblemID : int  - ProblemDescription : String  - CountOfOccurance : int
+ getProblemID() : int  + getProblemDescription() : String  + getCountOfOccurance : int  + setCountOfOccurance() : void

MAIN
- Problems : List<SELECTION_PROBLEM>
+ Burası derya deniz

TRANSCRIPT
- StudentID : int  - GPA : float  - TakenCourses : List<TAKEN_COURSE>  - TakenCredit : float  - CompletedCredit : float
+ getStudentID() : int  + getGPA() : float  + getTakenCourses() : List<TAKEN_COURSE>  +getTakenCredit() : float  + getCompleted Credit() : float  + setGPA () : void  + setTakenCredit : void  + setCompletedCredit() : void  + addTakenCourse(Course : TAKEN_COURSE) : void  + removeTakenCourse(Course : TAKEN_COURSE) : void

Tartışma Konuları

- > Ön koşul dersi normalde 1'den fazla değil. Liste şeklinde olmasındansa tek bir 'COURSE' olsun mu? Yukarıda COURSE şeklinde.
- > StudentName ne şekilde olsun? StudentName : String \*\*\*\* StudentName : List<String> \*\*\*\* StudentName : String, StudentMiddleName : String \*\*\* Ya da direkt Person classı ekleyip orda first, middle last name şeklinde alabiliriz. Yukarıda bu şekilde.. Hem bu sayede ilerde instructor class ını şimdiden düşünmüş oluruz.
- > int yerine daha küçük veri tiplerini kullanalım mı? (Bence kullanalım) örnek : Semester : int yerine Byte olabilir.
- > Ön koşul neden 3. dönemden başlıyor? İkinci dönemde de var normalde.
- > Quota ve CourseType ilişkisi kurulsun mu? Örnek : her TE 50 kişilik olabilir. Course adı altında CourseType = COURSE\_TYPE diyip işin içinden çıkabiliriz.
- > Must course'larda sınıf mevcudu aşılması durumunda yeni session açmamız gerekiyorsa ve bunu dinamik olarak ayarlamak istiyorsak 'Takvim' class'ı açalım mı?
- > Person class'ı kalsın mı ?
- > Hata class' ı kalsın mı?
- > Classroom class'ı kalsın mı?